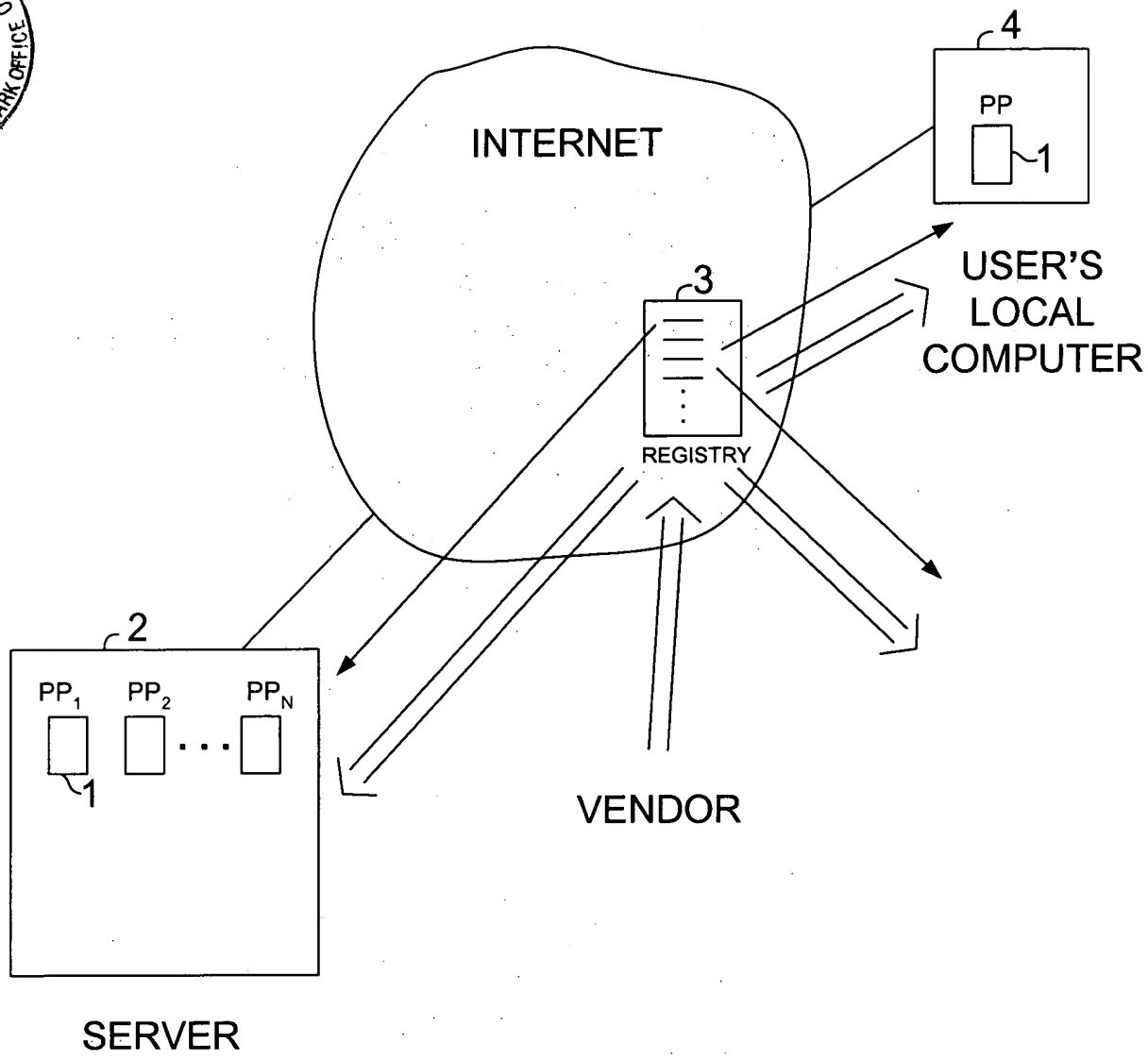
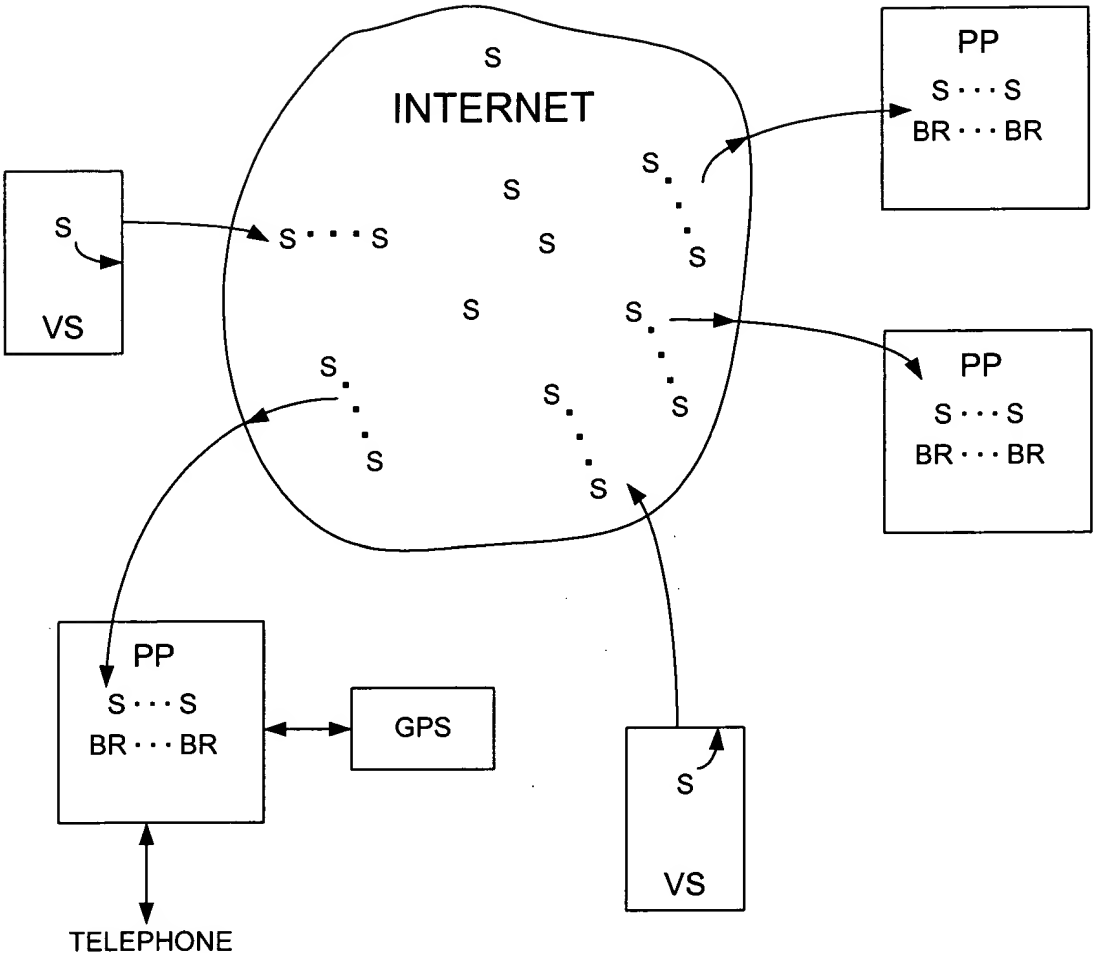


PP = PERSONAL PAGE
→ INDICATES NETWORK ACCESS
⇒ INDICATES SOFTWARE ACCESS

FIG. 1A



PP = PERSONAL PAGE
VS = VENDOR SITE
S = VENDOR SCRIPT
BR = BIDDING RULE

FIG. 1B

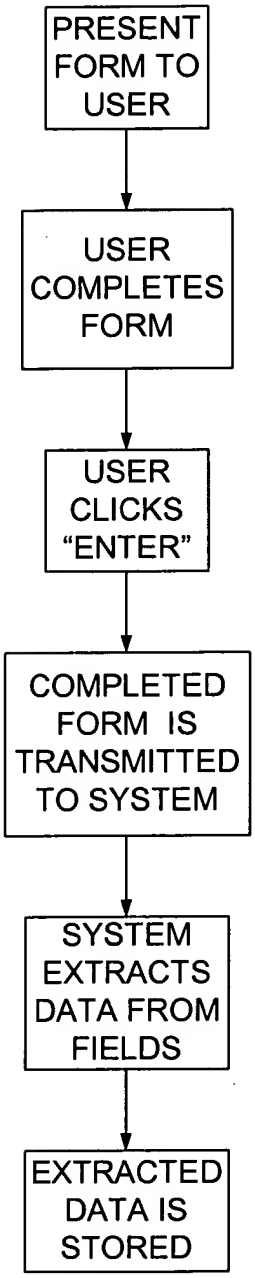

PRESENT FORM TO USER
USER COMPLETES FORM
USER CLICKS "ENTER"
COMPLETED FORM IS TRANSMITTED TO SYSTEM
SYSTEM EXTRACTS DATA FROM FIELDS
EXTRACTED DATA IS STORED

FIG. 2

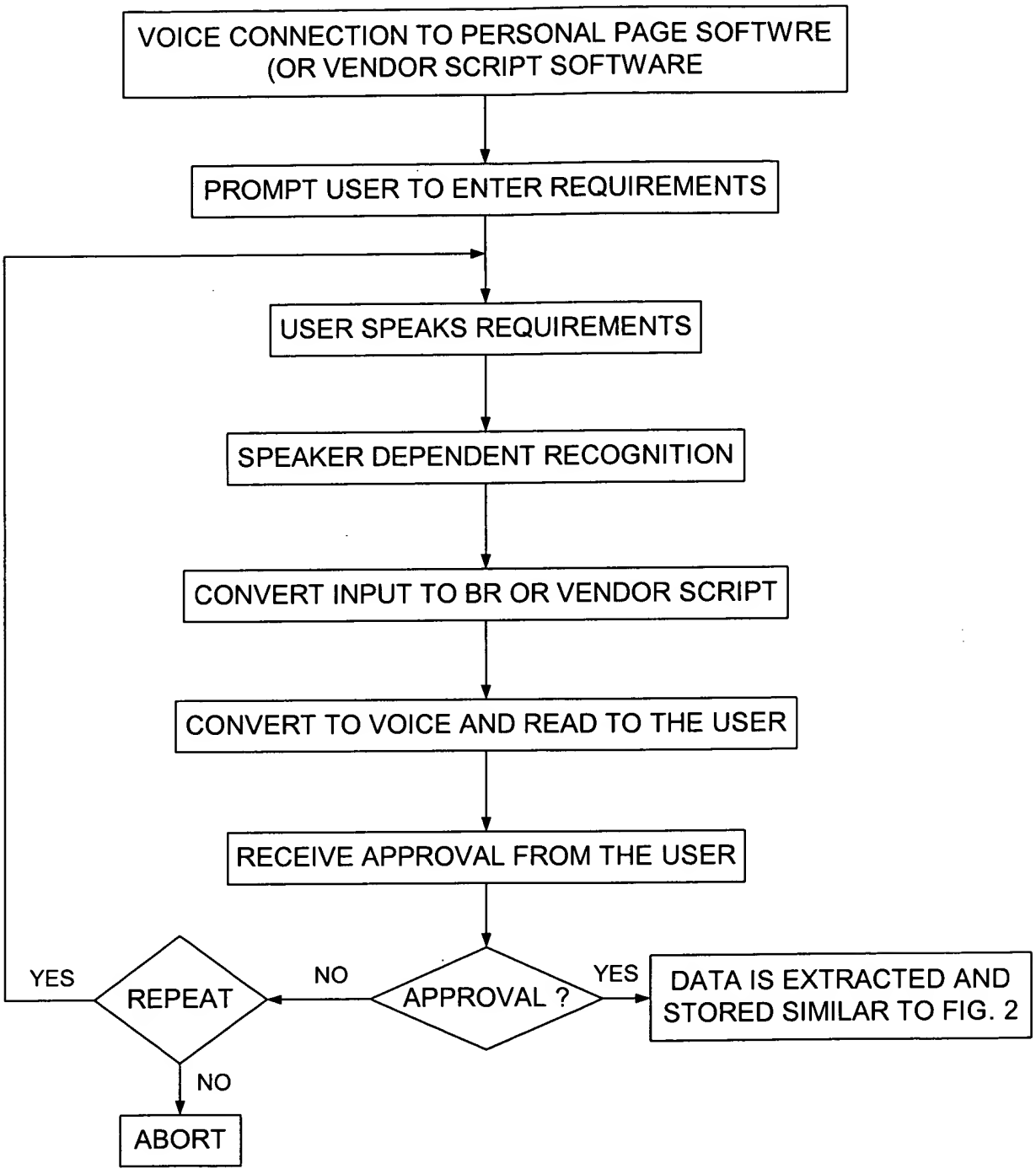

VOICE CONNECTION TO PERSONAL PAGE SOFTWRE (OR VENDOR SCRIPT SOFTWARE
PROMPT USER TO ENTER REQUIREMENTS
USER SPEAKS REQUIREMENTS
SPEAKER DEPENDENT RECOGNITION
CONVERT INPUT TO BR OR VENDOR SCRIPT
CONVERT TO VOICE AND READ TO THE USER
RECEIVE APPROVAL FROM THE USER
APPROVAL ?
YES
DATA IS EXTRACTED AND STORED SIMILAR TO FIG. 2
NO
REPEAT
YES
NO
ABORT

FIG. 3

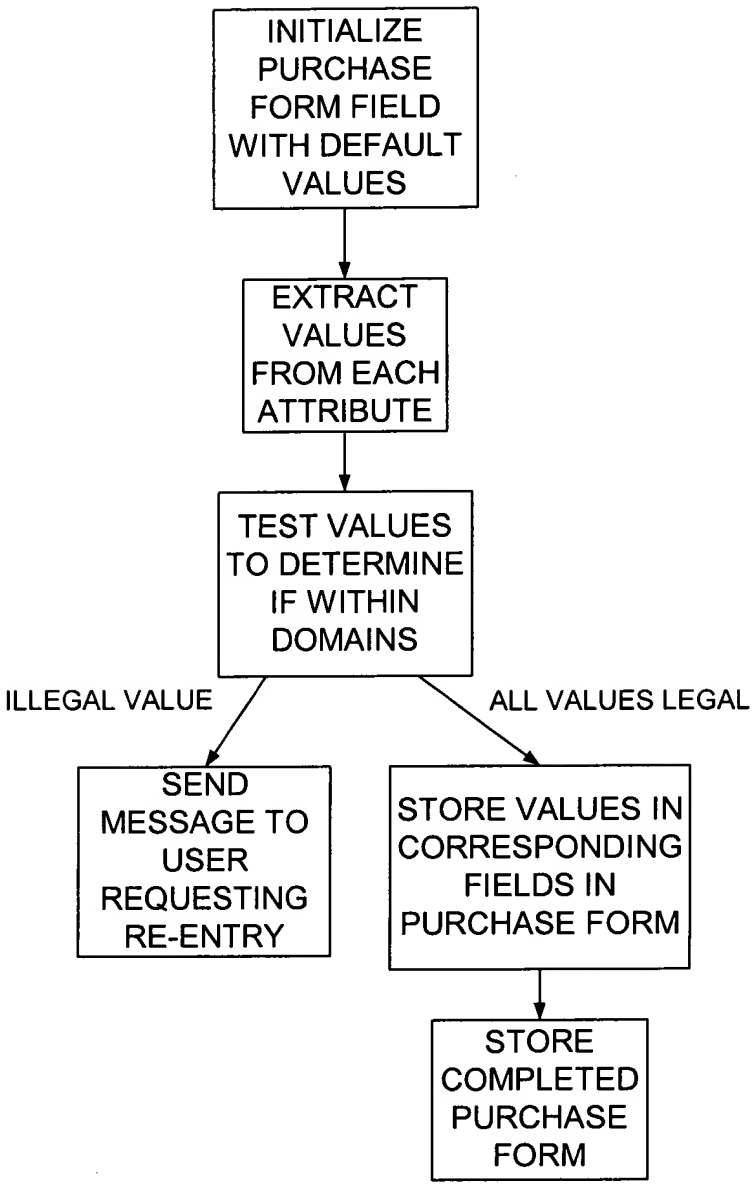

INITIALIZE PURCHASE FORM FIELD WITH DEFAULT VALUES
EXTRACT VALUES FROM EACH ATTRIBUTE
TEST VALUES TO DETERMINE IF WITHIN DOMAINS
ILLEGAL VALUE
ALL VALUES LEGAL
SEND MESSAGE TO USER REQUESTING RE-ENTRY
STORE VALUES IN CORRESPONDING FIELDS IN PURCHASE FORM
STORE COMPLETED PURCHASE FORM

FIG. 4

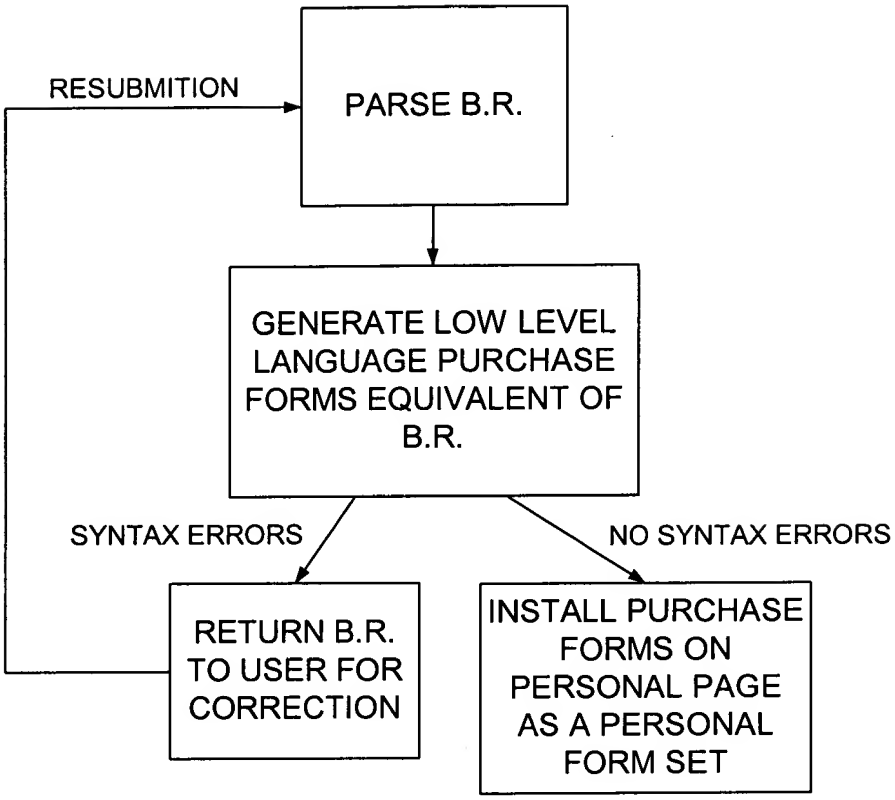

RESUBMITION
PARSE B.R.
GENERATE LOW LEVEL LANGUAGE PURCHASE FORMS EQUIVALENT OF B.R.
SYNTAX ERRORS
NO SYNTAX ERRORS
RETURN B.R. TO USER FOR CORRECTION
INSTALL PURCHASE FORMS ON PERSONAL PAGE AS A PERSONAL FORM SET

FIG. 5

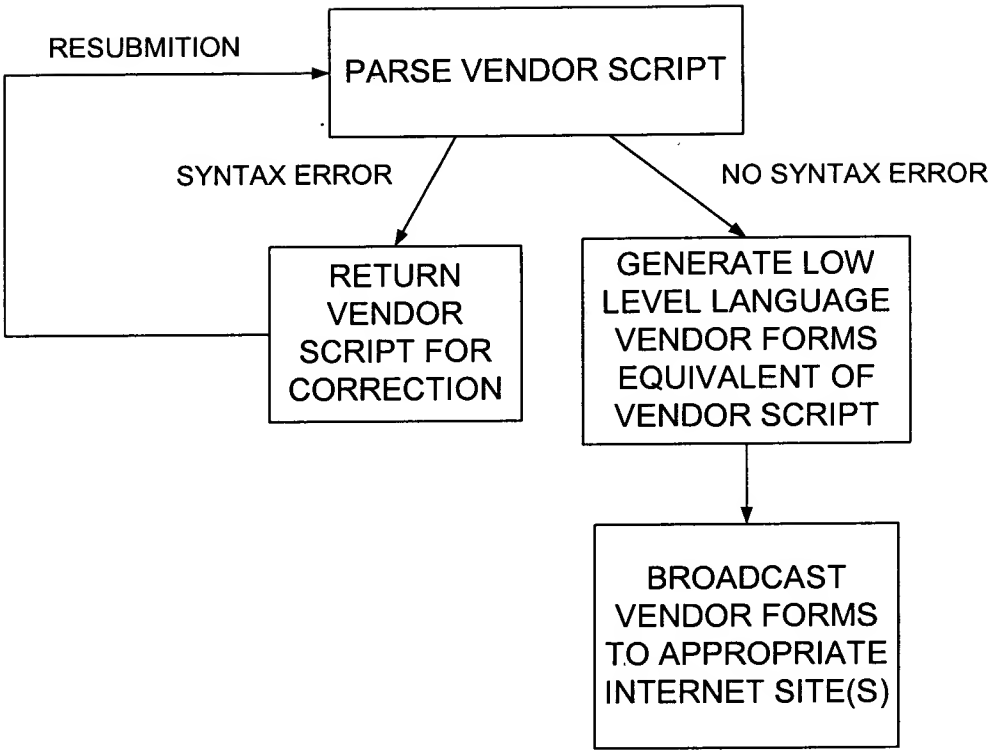

RESUBMITION
PARSE VENDOR SCRIPT
SYNTAX ERROR
NO SYNTAX ERROR
RETURN VENDOR SCRIPT FOR CORRECTION
GENERATE LOW LEVEL LANGUAGE VENDOR FORMS EQUIVALENT OF VENDOR SCRIPT
BROADCAST VENDOR FORMS TO APPROPRIATE INTERNET SITE(S)

FIG. 6

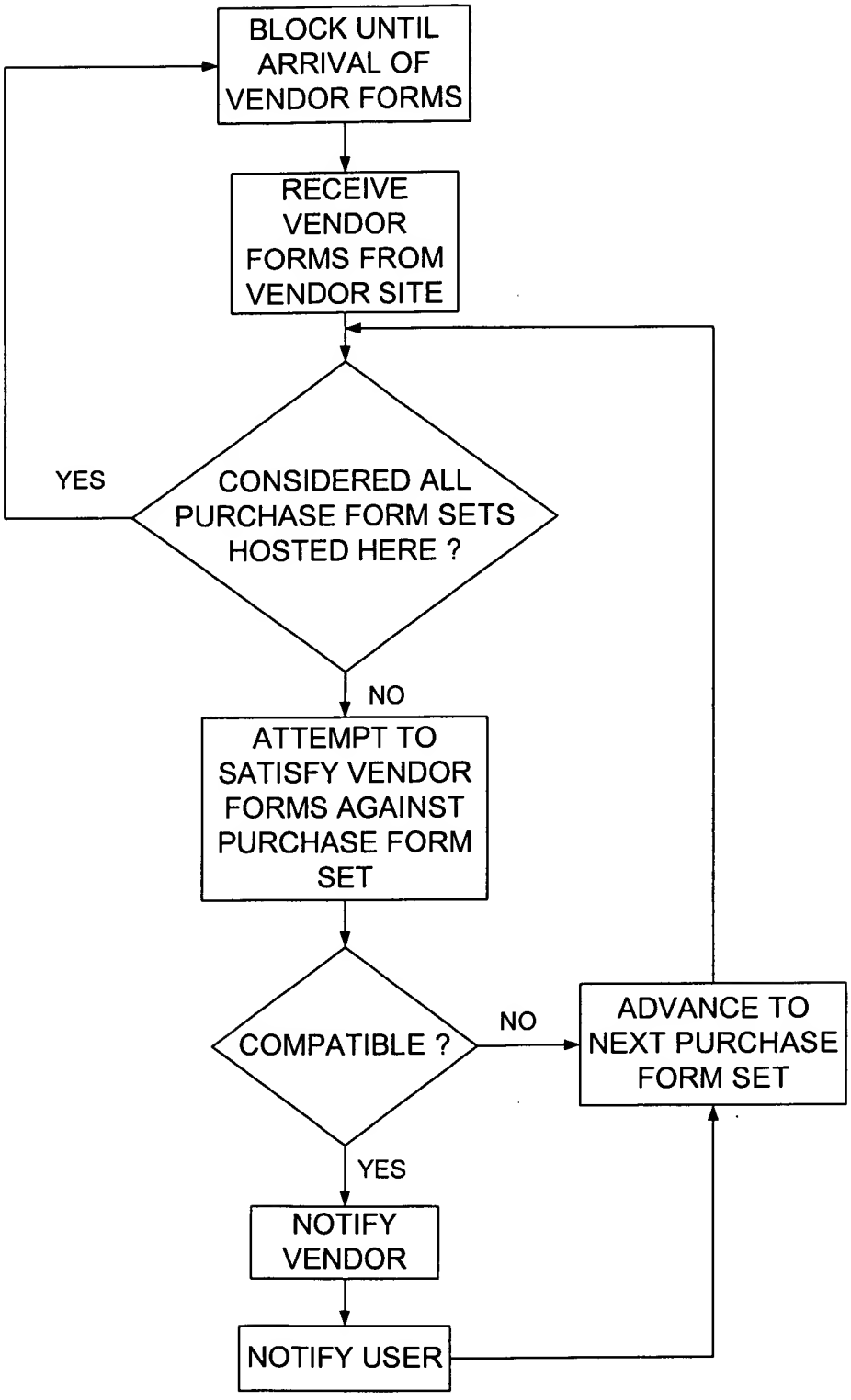

BLOCK UNTIL ARRIVAL OF VENDOR FORMS
RECEIVE VENDOR FORMS FROM VENDOR SITE
CONSIDERED ALL PURCHASE FORM SETS HOSTED HERE ?
YES
NO
ATTEMPT TO SATISFY VENDOR FORMS AGAINST PURCHASE FORM SET
COMPATIBLE ?
NO
ADVANCE TO NEXT PURCHASE FORM SET
YES
NOTIFY VENDOR
NOTIFY USER

FIG. 7

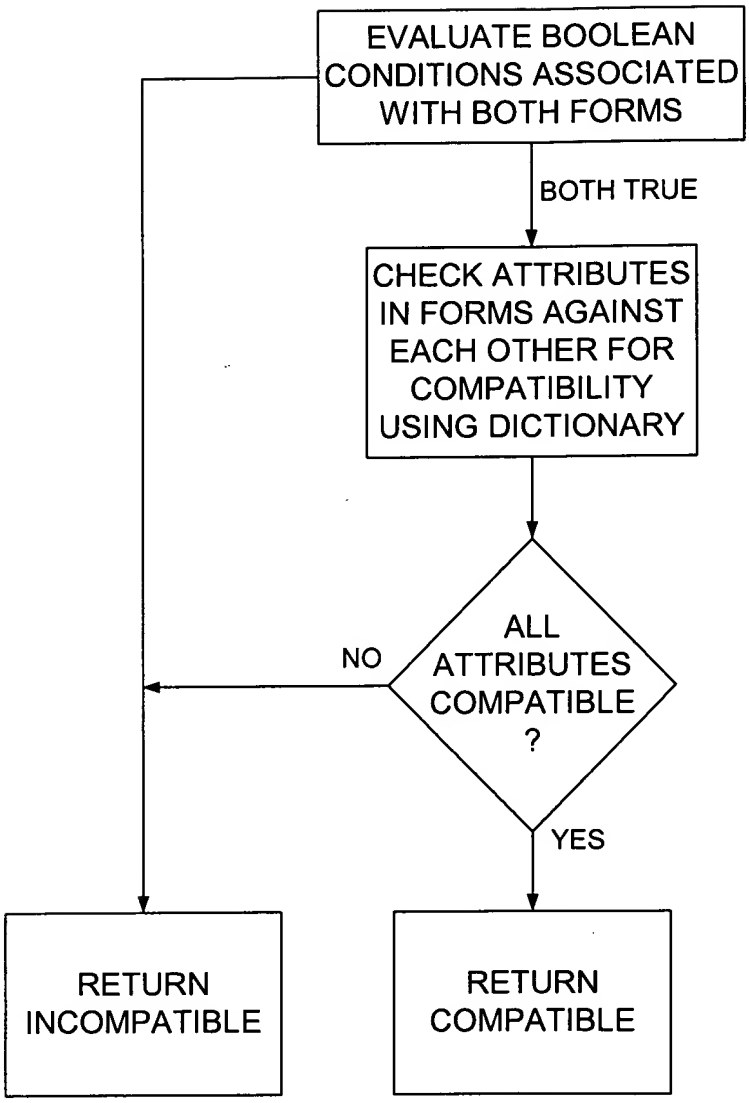

EVALUATE BOOLEAN CONDITIONS ASSOCIATED WITH BOTH FORMS
BOTH TRUE
CHECK ATTRIBUTES IN FORMS AGAINST EACH OTHER FOR COMPATIBILITY USING DICTIONARY
ALL ATTRIBUTES COMPATIBLE ?
NO
YES
RETURN INCOMPATIBLE
RETURN COMPATIBLE

FIG. 8

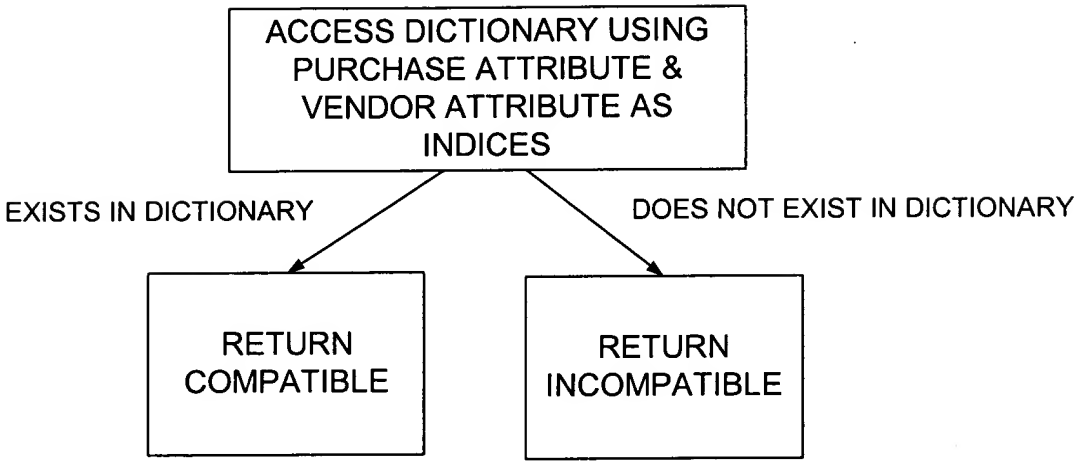

ACCESS DICTIONARY USING PURCHASE ATTRIBUTE & VENDOR ATTRIBUTE AS INDICES
EXISTS IN DICTIONARY
DOES NOT EXIST IN DICTIONARY
RETURN COMPATIBLE
RETURN INCOMPATIBLE

FIG. 9

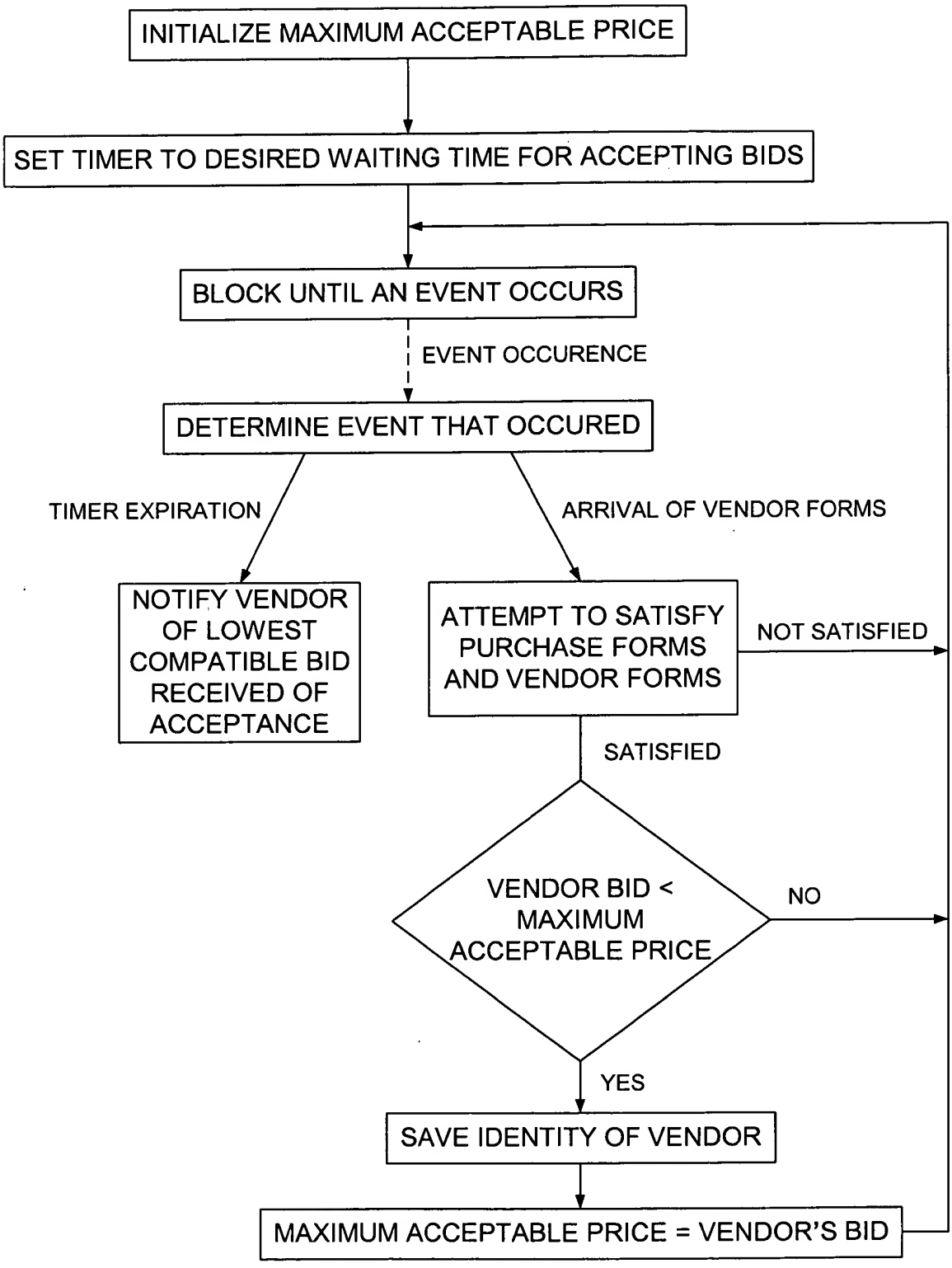

INITIALIZE MAXIMUM ACCEPTABLE PRICE
SET TIMER TO DESIRED WAITING TIME FOR ACCEPTING BIDS
BLOCK UNTIL AN EVENT OCCURS
EVENT OCCURENCE
DETERMINE EVENT THAT OCCURED
TIMER EXPIRATION
ARRIVAL OF VENDOR FORMS
NOTIFY VENDOR OF LOWEST COMPATIBLE BID RECEIVED OF ACCEPTANCE
ATTEMPT TO SATISFY PURCHASE FORMS AND VENDOR FORMS
NOT SATISFIED
SATISFIED
VENDOR BID < MAXIMUM ACCEPTABLE PRICE
NO
YES
SAVE IDENTITY OF VENDOR
MAXIMUM ACCEPTABLE PRICE = VENDOR'S BID

FIG. 10

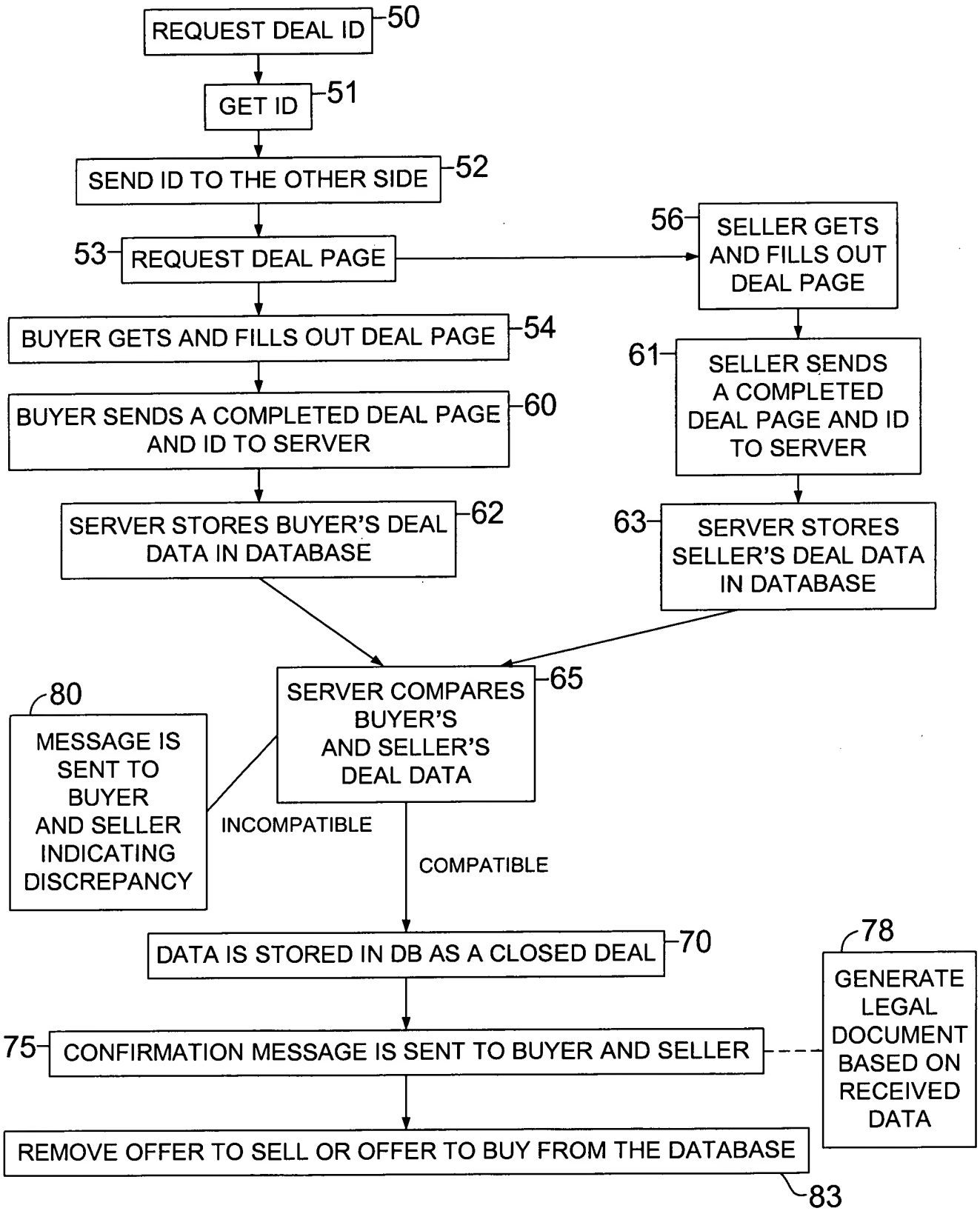

REQUEST DEAL ID
50
GET ID
51
SEND ID TO THE OTHER SIDE
52
53
REQUEST DEAL PAGE
56
SELLER GETS AND FILLS OUT DEAL PAGE
BUYER GETS AND FILLS OUT DEAL PAGE
54
61
SELLER SENDS A COMPLETED DEAL PAGE AND ID TO SERVER
BUYER SENDS A COMPLETED DEAL PAGE AND ID TO SERVER
60
SERVER STORES BUYER'S DEAL DATA IN DATABASE
62
63
SERVER STORES SELLER'S DEAL DATA IN DATABASE
SERVER COMPARES BUYER'S AND SELLER'S DEAL DATA
65
80
MESSAGE IS SENT TO BUYER AND SELLER INDICATING DISCREPANCY
INCOMPATIBLE
COMPATIBLE
DATA IS STORED IN DB AS A CLOSED DEAL
70
78
GENERATE LEGAL DOCUMENT BASED ON RECEIVED DATA
75
CONFIRMATION MESSAGE IS SENT TO BUYER AND SELLER
REMOVE OFFER TO SELL OR OFFER TO BUY FROM THE DATABASE
83

FIG. 11

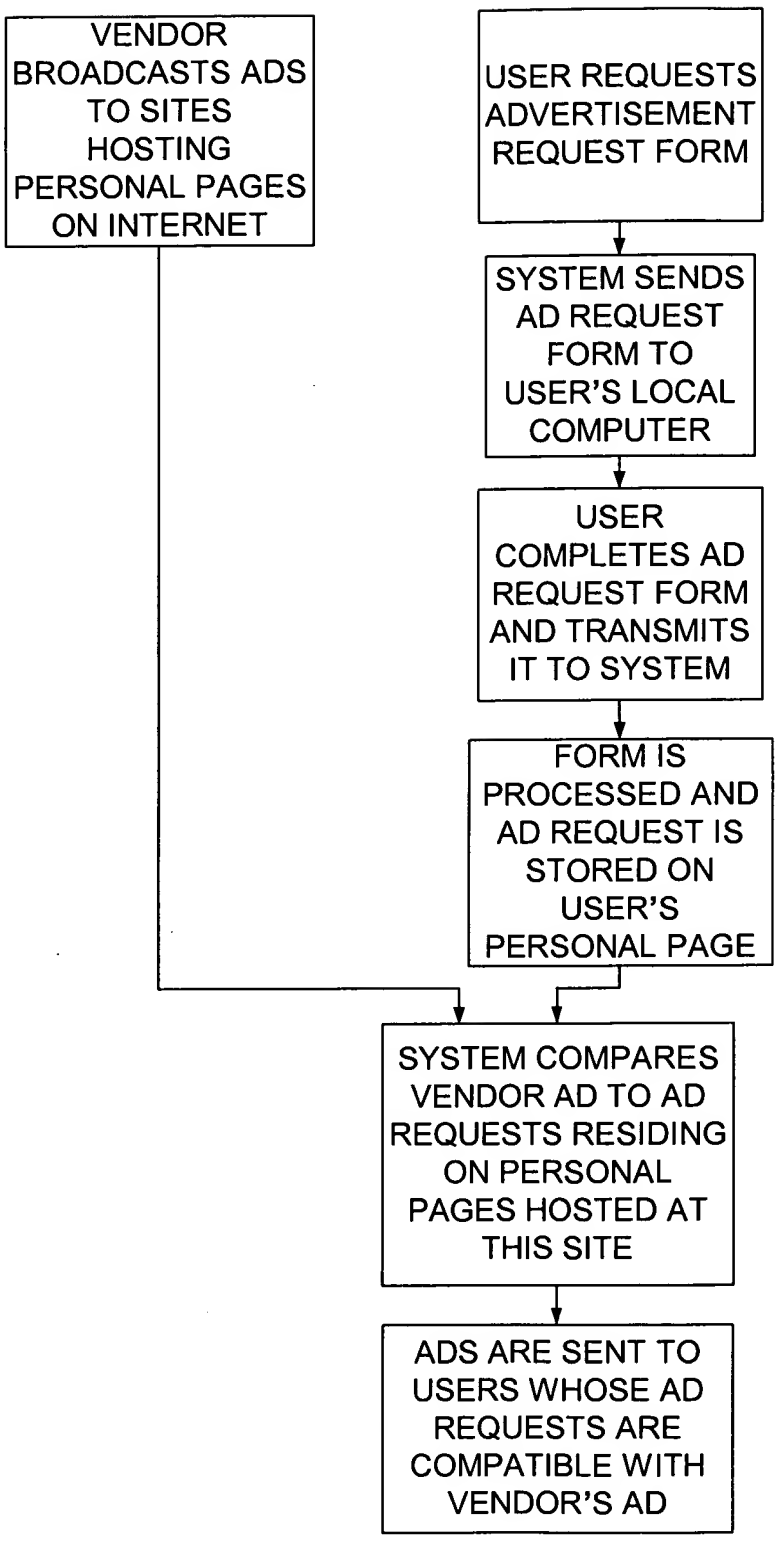

VENDOR BROADCASTS ADS TO SITES HOSTING PERSONAL PAGES ON INTERNET
USER REQUESTS ADVERTISEMENT REQUEST FORM
SYSTEM SENDS AD REQUEST FORM TO USER'S LOCAL COMPUTER
USER COMPLETES AD REQUEST FORM AND TRANSMITS IT TO SYSTEM
FORM IS PROCESSED AND AD REQUEST IS STORED ON USER'S PERSONAL PAGE
SYSTEM COMPARES VENDOR AD TO AD REQUESTS RESIDING ON PERSONAL PAGES HOSTED AT THIS SITE
ADS ARE SENT TO USERS WHOSE AD REQUESTS ARE COMPATIBLE WITH VENDOR'S AD

FIG. 12

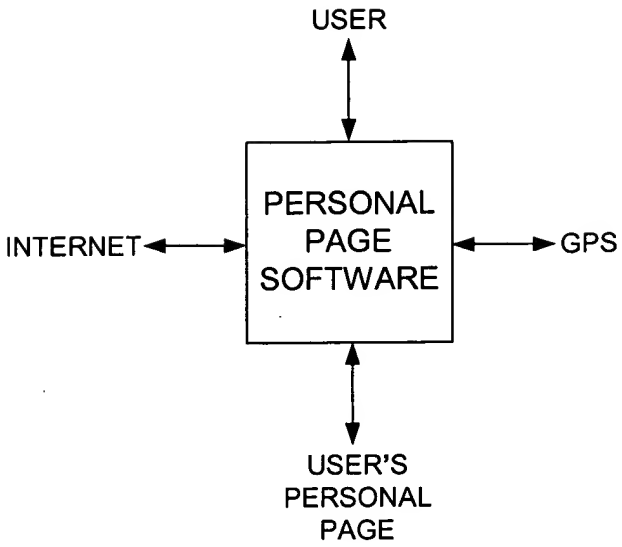


FIG. 13

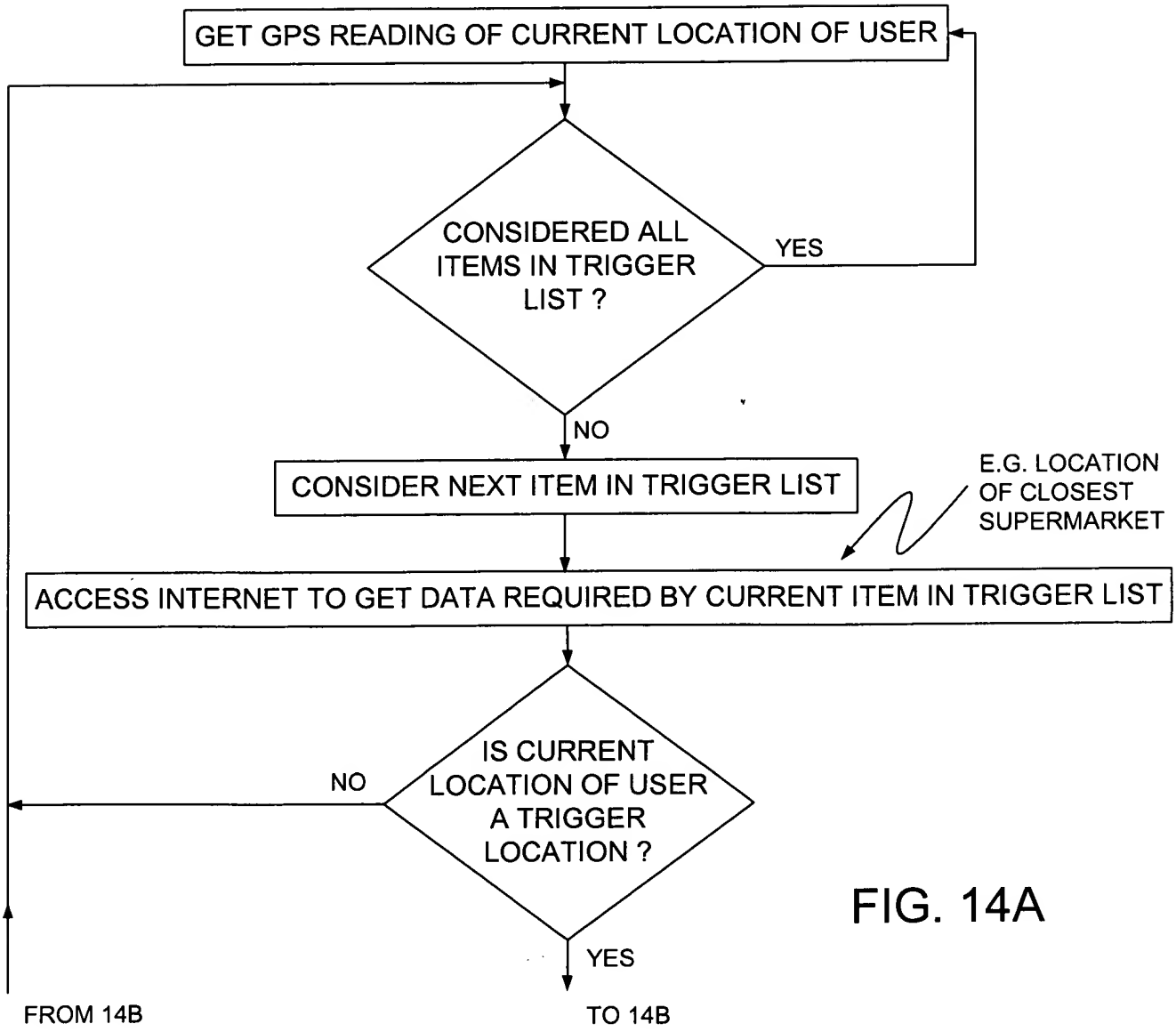


FIG. 14A

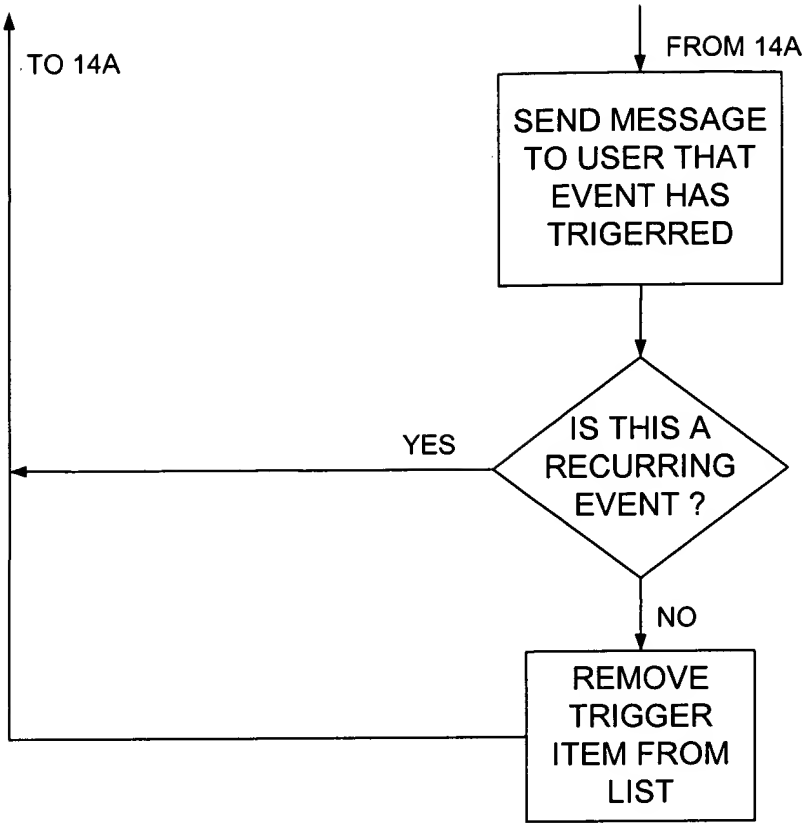

TO 14A
FROM 14A
SEND MESSAGE TO USER THAT EVENT HAS TRIGERRED
IS THIS A RECURRING EVENT ?
YES
NO
REMOVE TRIGGER ITEM FROM LIST

FIG. 14B

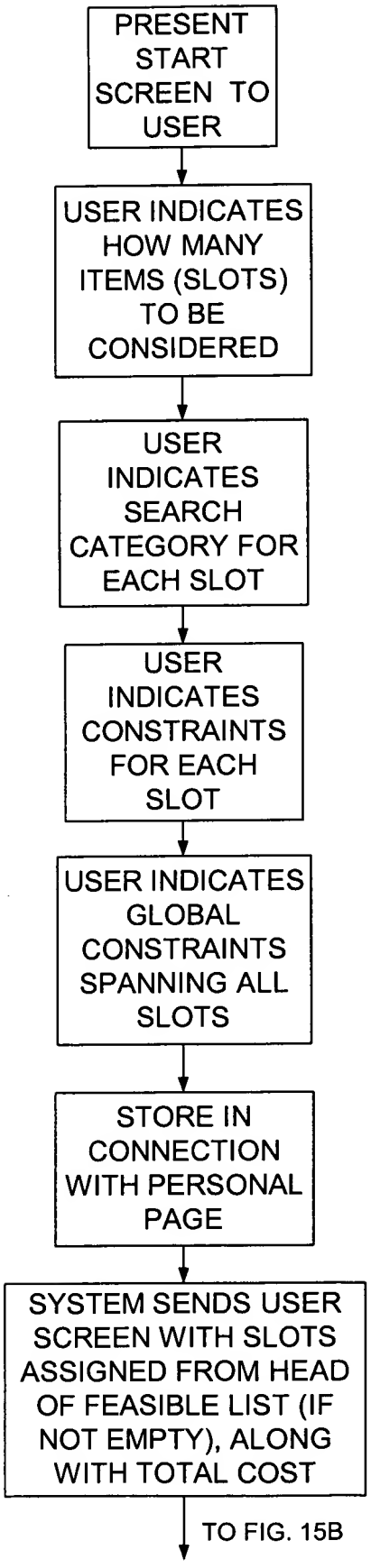

PRESENT START SCREEN TO USER
USER INDICATES HOW MANY ITEMS (SLOTS) TO BE CONSIDERED
USER INDICATES SEARCH CATEGORY FOR EACH SLOT
USER INDICATES CONSTRAINTS FOR EACH SLOT
USER INDICATES GLOBAL CONSTRAINTS SPANNING ALL SLOTS
STORE IN CONNECTION WITH PERSONAL PAGE
SYSTEM SENDS USER SCREEN WITH SLOTS ASSIGNED FROM HEAD OF FEASIBLE LIST (IF NOT EMPTY), ALONG WITH TOTAL COST
TO FIG. 15B

FIG. 15A

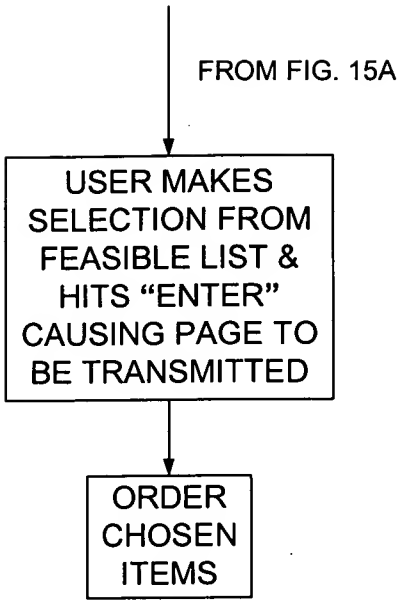

FROM FIG. 15A
USER MAKES SELECTION FROM FEASIBLE LIST & HITS "ENTER" CAUSING PAGE TO BE TRANSMITTED
ORDER CHOSEN ITEMS

FIG. 15B

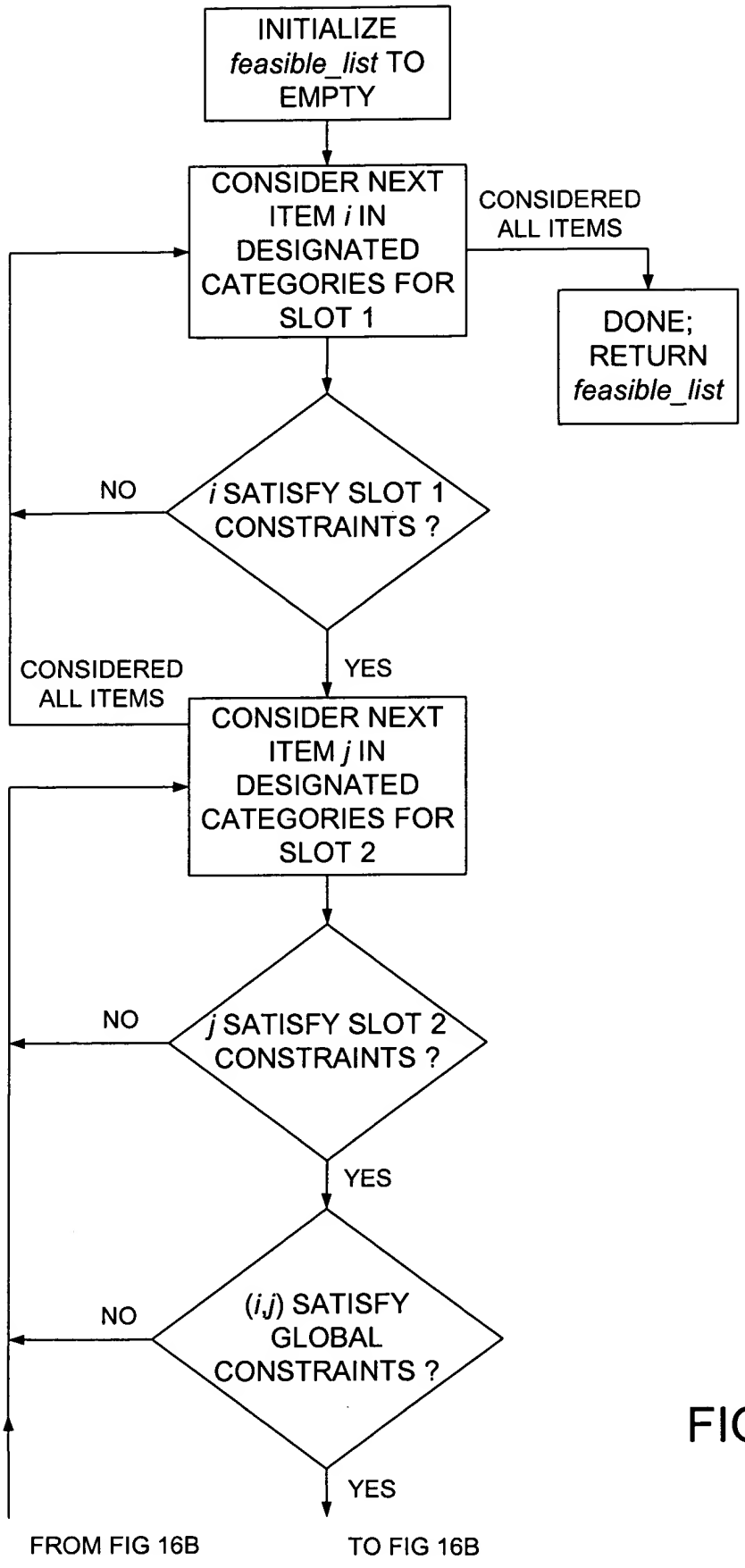

INITIALIZE *feasible_list* TO EMPTY
CONSIDER NEXT ITEM *i* IN DESIGNATED CATEGORIES FOR SLOT 1
CONSIDERED ALL ITEMS
DONE; RETURN *feasible_list*
i SATISFY SLOT 1 CONSTRAINTS ?
NO
YES
CONSIDERED ALL ITEMS
CONSIDER NEXT ITEM *j* IN DESIGNATED CATEGORIES FOR SLOT 2
j SATISFY SLOT 2 CONSTRAINTS ?
NO
YES
(*i*,*j*) SATISFY GLOBAL CONSTRAINTS ?
NO
YES
FROM FIG 16B
TO FIG 16B

FIG. 16A

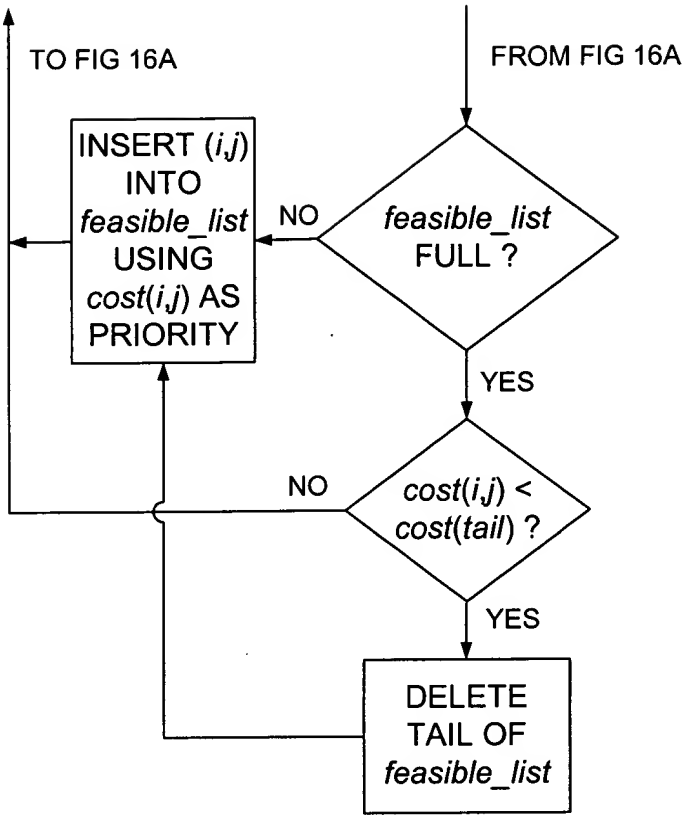

TO FIG 16A
FROM FIG 16A
INSERT (i,j) INTO feasible_list USING cost(i,j) AS PRIORITY
NO
feasible_list FULL ?
YES
NO
cost(i,j) < cost(tail) ?
YES
DELETE TAIL OF feasible_list

FIG. 16B

```
feasible_list = empty;          // priority queue; cost(a,b) is priority
for (i ranging over items in  its designated categories)
  if (i satisfies slot 1 constraints)
    for (j ranging over items in its designated categories)
      if (j satisfies slot 2 contraints)
        if ( (i,j) satisfies global contraints)
          if feasible_list not full
              insert (i,j) and cost(i,j) into feasible_list;
          else if ( cost(i,j) < cost(tail) )
            {
              delete tail from feasible_list;
              insert (i,j) and cost(i,j) into feasible_list;
            }
return (feasible_list);
```

FIG. 16C

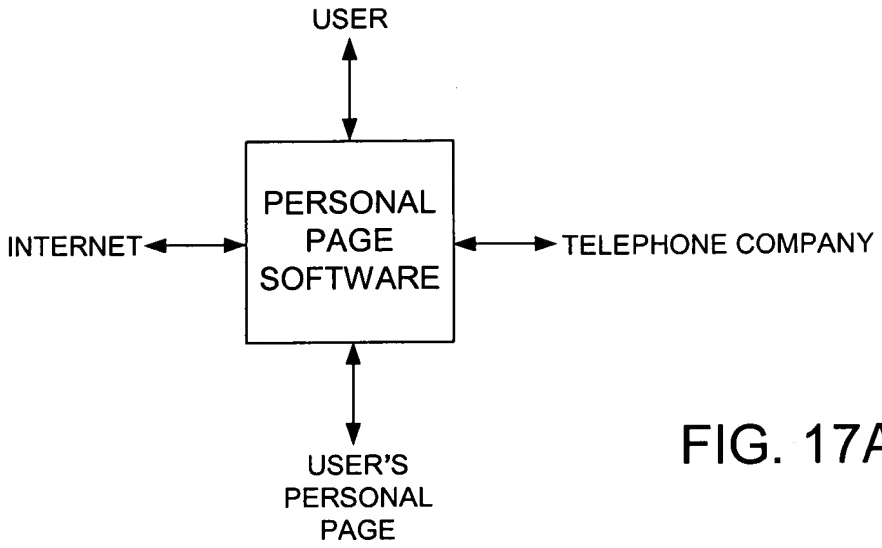


FIG. 17A

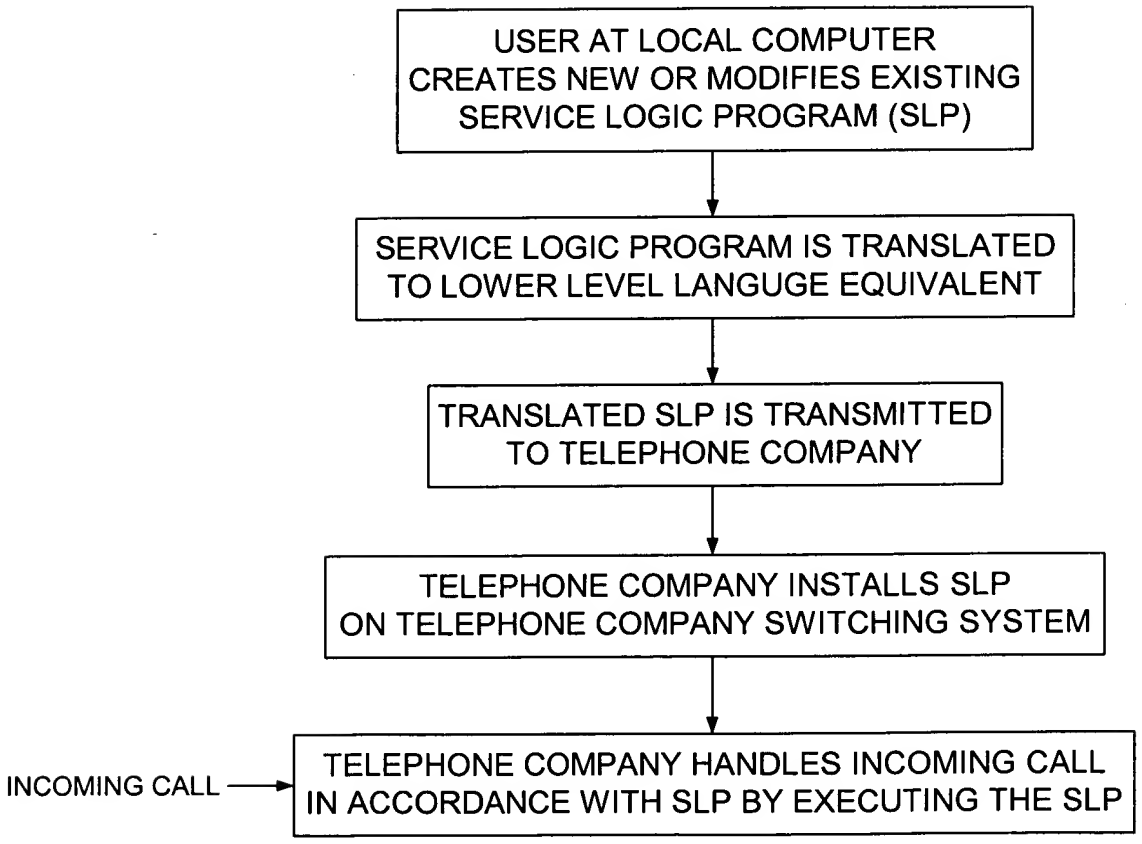


FIG. 17B

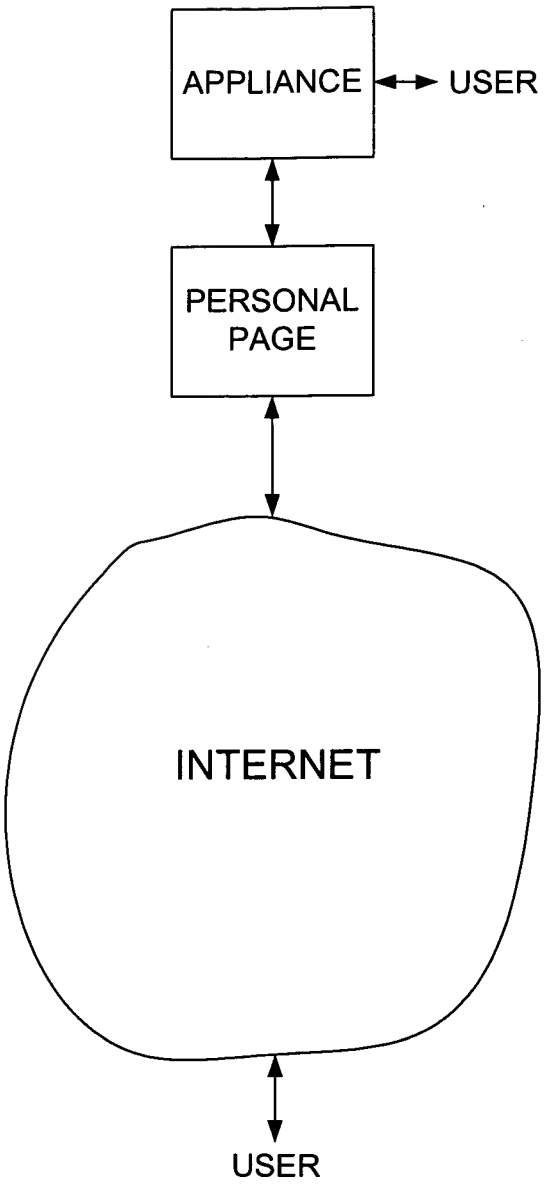


FIG. 18

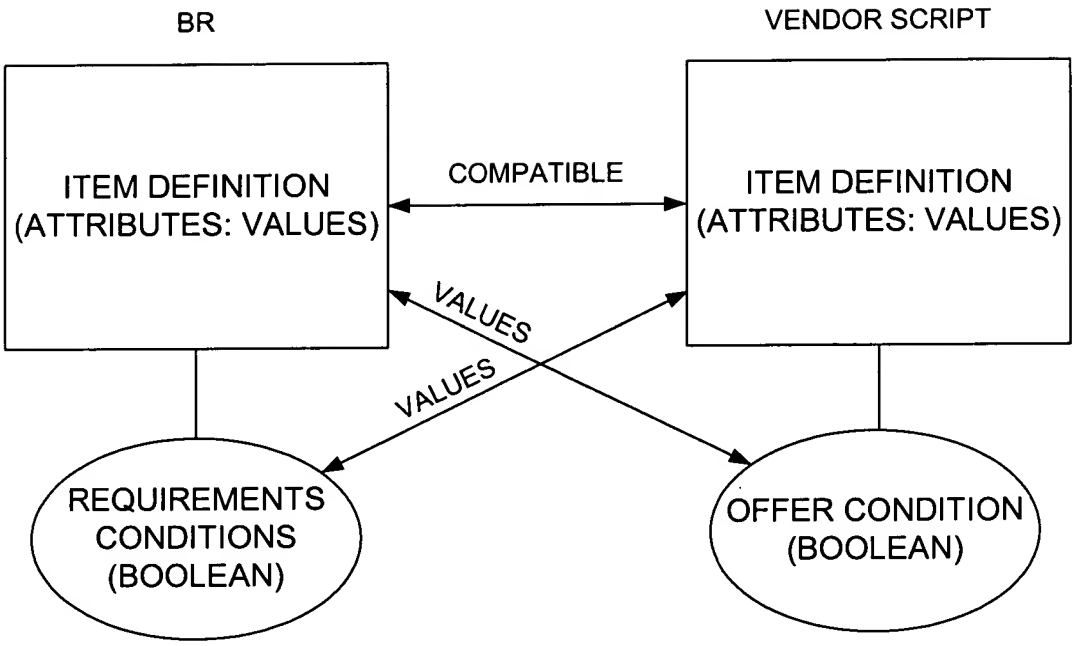

BR
VENDOR SCRIPT
ITEM DEFINITION
(ATTRIBUTES: VALUES)
COMPATIBLE
ITEM DEFINITION
(ATTRIBUTES: VALUES)
VALUES
VALUES
REQUIREMENTS
CONDITIONS
(BOOLEAN)
OFFER CONDITION
(BOOLEAN)

FIG. 19

{ ACCEPT_BR || ACCEPT_VENDOR_SCRIPT || MATCHMAKER
|| SERVICE_ACTIVATION_LIST || SERVICE_SCRIPT_LIST }

FIG. 20

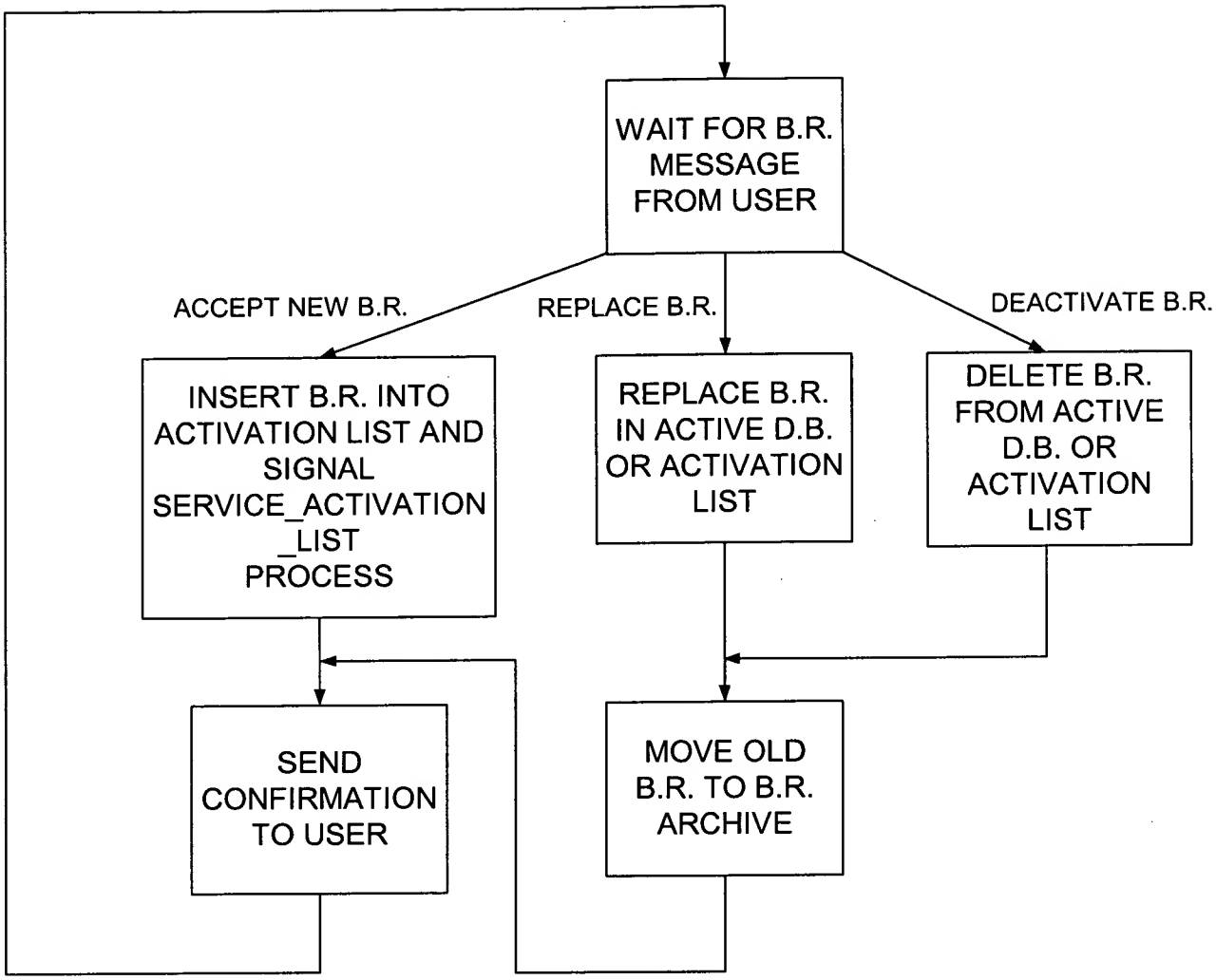

WAIT FOR B.R. MESSAGE FROM USER
ACCEPT NEW B.R.
REPLACE B.R.
DEACTIVATE B.R.
INSERT B.R. INTO ACTIVATION LIST AND SIGNAL SERVICE_ACTIVATION _LIST PROCESS
REPLACE B.R. IN ACTIVE D.B. OR ACTIVATION LIST
DELETE B.R. FROM ACTIVE D.B. OR ACTIVATION LIST
SEND CONFIRMATION TO USER
MOVE OLD B.R. TO B.R. ARCHIVE

FIG. 21

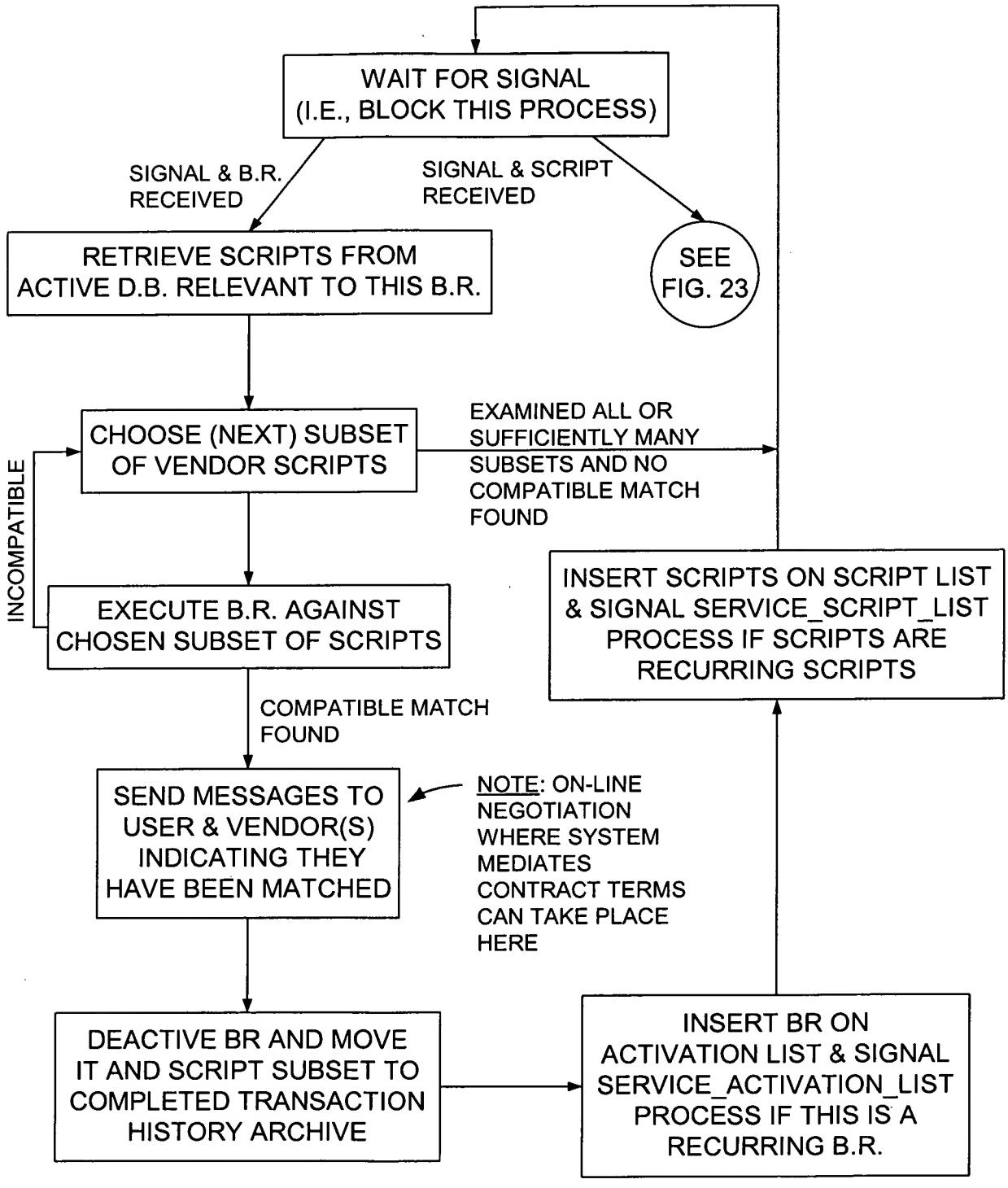

WAIT FOR SIGNAL
(I.E., BLOCK THIS PROCESS)
SIGNAL & B.R.
RECEIVED
SIGNAL & SCRIPT
RECEIVED
RETRIEVE SCRIPTS FROM
ACTIVE D.B. RELEVANT TO THIS B.R.
SEE
FIG. 23
CHOOSE (NEXT) SUBSET
OF VENDOR SCRIPTS
EXAMINED ALL OR
SUFFICIENTLY MANY
SUBSETS AND NO
COMPATIBLE MATCH
FOUND
INCOMPATIBLE
EXECUTE B.R. AGAINST
CHOSEN SUBSET OF SCRIPTS
INSERT SCRIPTS ON SCRIPT LIST
& SIGNAL SERVICE_SCRIPT_LIST
PROCESS IF SCRIPTS ARE
RECURRING SCRIPTS
COMPATIBLE MATCH
FOUND
SEND MESSAGES TO
USER & VENDOR(S)
INDICATING THEY
HAVE BEEN MATCHED
NOTE: ON-LINE
NEGOTIATION
WHERE SYSTEM
MEDIATES
CONTRACT TERMS
CAN TAKE PLACE
HERE
DEACTIVE BR AND MOVE
IT AND SCRIPT SUBSET TO
COMPLETED TRANSACTION
HISTORY ARCHIVE
INSERT BR ON
ACTIVATION LIST & SIGNAL
SERVICE_ACTIVATION_LIST
PROCESS IF THIS IS A
RECURRING B.R.

FIG. 22

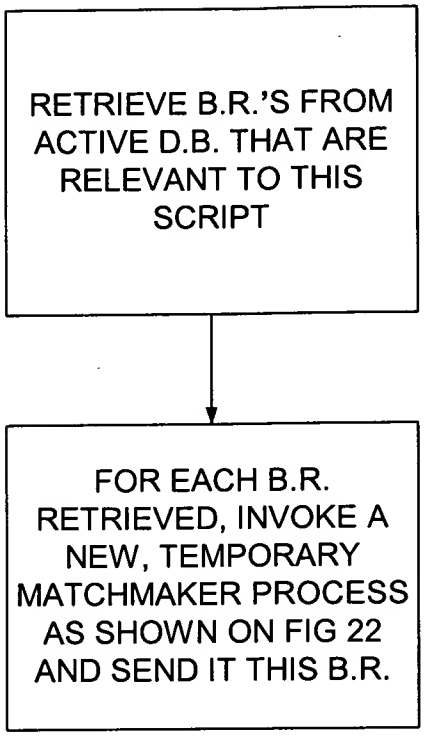

RETRIEVE B.R.'S FROM ACTIVE D.B. THAT ARE RELEVANT TO THIS SCRIPT
FOR EACH B.R. RETRIEVED, INVOKE A NEW, TEMPORARY MATCHMAKER PROCESS AS SHOWN ON FIG 22 AND SEND IT THIS B.R.

FIG. 23

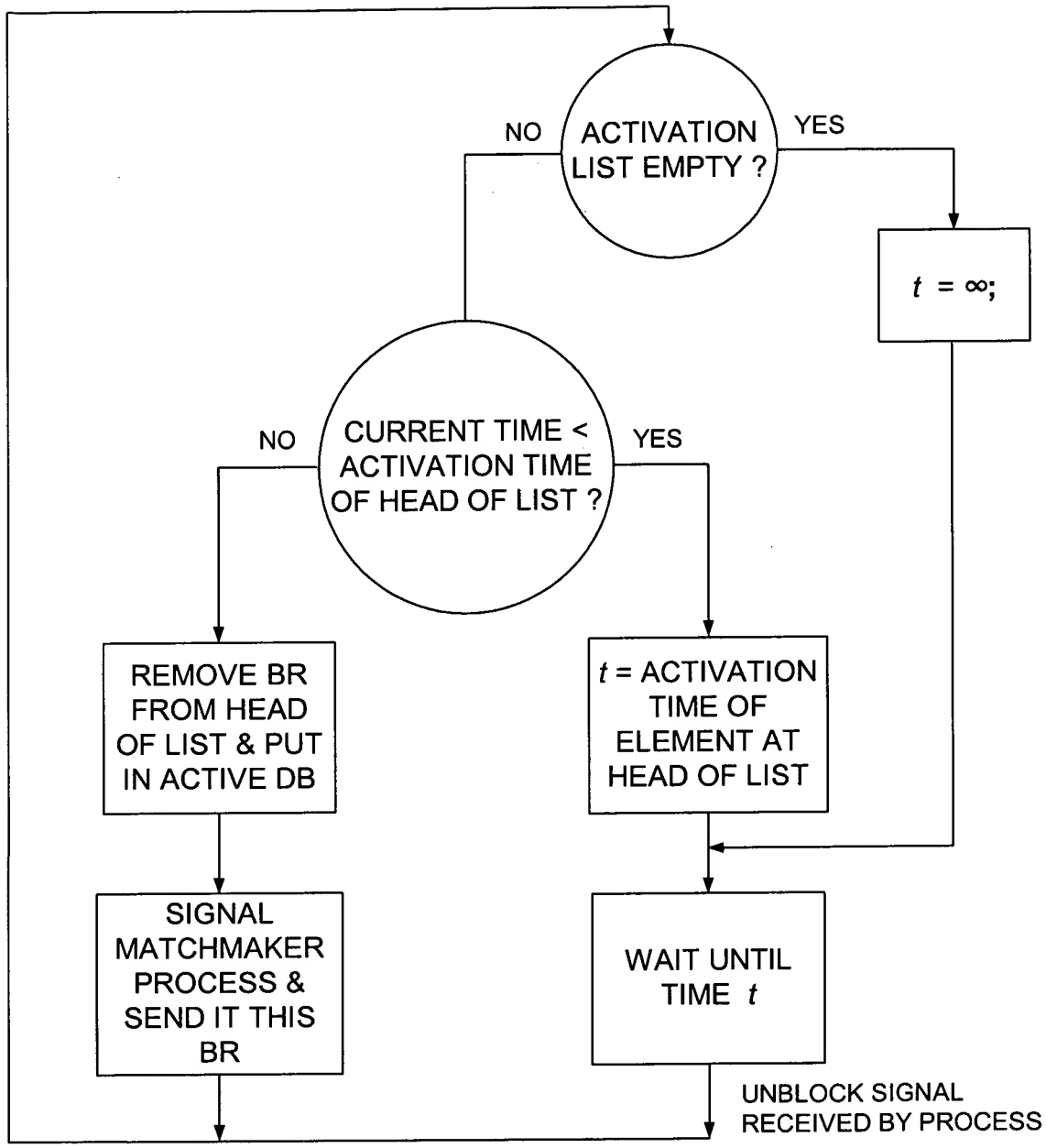

ACTIVATION LIST EMPTY ?
NO
YES
t = ∞;
CURRENT TIME < ACTIVATION TIME OF HEAD OF LIST ?
NO
YES
REMOVE BR FROM HEAD OF LIST & PUT IN ACTIVE DB
t = ACTIVATION TIME OF ELEMENT AT HEAD OF LIST
SIGNAL MATCHMAKER PROCESS & SEND IT THIS BR
WAIT UNTIL TIME t
UNBLOCK SIGNAL RECEIVED BY PROCESS

FIG. 24